AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

LISTING OF CLAIMS:

1. (Currently Amended) A cryptographic method during which an integer division of the type q = a div b and r = a mod b is performed, ~~with~~ where q is a quotient, a is a number ~~of~~ containing m bits, b is a number ~~of~~ containing n bits, with n less than or equal to m and b_{n-1} is non-zero, b_{n-1} being the most significant bit of b, ~~a method during which, at each iteration of a loop subscripted by i varying between 1 and m-n+1,~~ comprising the following steps:

(i) performing a partial division of a word A, comprising ~~of~~ n bits of the number a, by the number b ~~is performed in order~~ to obtain a bit of the quotient q, wherein at least one of the numbers a and b comprises secret data;

~~the method being characterised in that~~ (ii) repeating step (i) for m-n+1 iterations with the same operations ~~are~~ being performed at each iteration, ~~whatever~~ regardless of the value of the quotient bit obtained, to obtain the quotient q; and

(iii) generating encrypted or decrypted data in accordance with said quotient.

2. (Currently Amended) A method according to Claim 1, ~~during which~~ wherein, at each iteration, an addition of the number b to the word A and a subtraction of the number b from the word A are performed.

3. (Currently Amended) A method according to ~~one of Claims 1 to 2, during which~~ claim 1, wherein all the following steps are performed :

Input : $a = (0, a_{m-1}, \ldots, a_0)$
$b = (b_{n-1}, \ldots, b_0)$

Output: $q = a \text{ div } b$ and $r = a \bmod b$

$A = (0, a_{m-1}, \ldots, a_{m-n+1})$; $\sigma' <- 1$

For $j = 1$ to $(m-n+1)$, do:

$a <- SHL_{m+1}(a, 1)$; $\sigma <-$ carry

$A <- (\sigma')SUB_n(A, b) + (\neg\sigma')ADD_n(A, b)$

$\sigma <-(\sigma'$ AND $\sigma') / (\sigma'$ AND carry$)/ (\sigma'$ AND carry$)$

lsb(a) σ'

$\sigma' <- \sigma$

End For

if $(\neg\sigma = TRUE)$ then $A <- ADD_n(A, b)$

4. (Currently Amended) A method according to Claim 1~~, during which~~ wherein, at each iteration, ~~an operation of addition~~ either ~~of~~ the number b or of a number $\bar{b}$ complementary to the number b ~~with~~ is added to the word A ~~is performed~~.

5. (Currently Amended) A method according to Claim 4, ~~during which~~ further including the step, at each iteration, ~~an~~ of updating ~~is also carried out of~~ a first variable (σ') indicating whether, during the following iteration, the number b or the number $\bar{b}$ ~~must~~ is to be added with the word A according to the quotient bit produced ~~(lsb(a))~~.

6. (Currently Amended) A method according to Claim 4 ~~or Claim 5~~, ~~during which~~ wherein all the following steps are performed :

Input : $a = (0, a_{m-1}, \ldots, a_0)$

$b = (b_{n-1}, \ldots, b_0)$

Output: $q = a \text{ div } b$ and $r = a \bmod b$

$A = (0, a_{m-1}, \ldots, a_{m-n+1})$; $\sigma' <- 1$; $\bar{b} <- CPL2_N(b)$

For $j = 1$ to $(m-n+1)$, do:

$a <- SHL_{m+1}(a, 1)$; $\sigma <-$ carry

$d_{addr} <- b_{addr} + \sigma' (\bar{b}_{addr} - b_{addr})$

$A <- ADD_n(A, d)$

$\sigma <-(\sigma'$ AND $\sigma') / (\sigma'$ AND carry$)/ (\sigma'$ AND carry$)$

$lsb(a) <- \sigma'$

$\sigma' <- \sigma$

End For

if $(\neg\sigma = TRUE)$ then $A <- ADD_n(A, b)$

7. (Currently Amended) A method according to Claim 1, ~~during which~~ further including the steps, at each iteration, of performing an operation of complement to 2^n of an updated data item (b or $\bar{b}$) or of a notional data item (c or $\bar{c}$) ~~is performed~~, and ~~then an operation of addition of~~ adding the updated data item with the word A.

8. (Currently Amended) A method according to Claim 7, ~~during which~~ further including the step, at each iteration, ~~an operation~~ of updating a second variable (δ) ~~is also performed~~, indicating whether, during the following iteration, the operation of complement to 2^n ~~must~~ is to be performed on the updated data item or on the notional data item.

9. (Currently Amended) A method according to ~~one of Claims 7 or 8, in which~~ claim 7, further including the step, at each iteration, ~~there is also performed an~~ of updating ~~of~~ a third variable (β) indicating whether the updated data item is equal to the data item b or to its complement to 2^n.

10. (Currently Amended) A method according to ~~one of Claims 7 to 9, during which~~ claim 7, wherein all the following steps are also performed :

Input : $a = (0, a_{m-1}, \ldots, a_0)$

$b = (b_{n-1}, \ldots, b_0)$

Output: $q = a \text{ div } b$ and $r = a \bmod b$

$\sigma' <- 1$; $\beta <- 1, \gamma <- 1$; $A = (0, a_{m-1}, \ldots., a_{m-n+1})$

for $j = 1$ to $(m-n+1)$, do:

$a <- SHL_{m+1}(a, 1)$; $\sigma <-$ carry

$\delta <- \sigma' / \beta$

$d_{addr} <- b_{addr} + \delta (c_{addr} - b_{addr})$

$d <- CPL2_n(d)$

A <- ADD_n(A, b)

σ <-(σ AND σ') / (σ AND carry)/ (σ' AND carry)

β <- ¬σ' ; γ <- γ / δ; σ' <- σ

lsb(a) = σ

end for

if (¬σ = TRUE) then A <- ADD_n(A, b)

11. (Currently Amended) A method according to Claim 10, ~~during which~~ wherein, at the end, the following operations are performed :

if (¬β = TRUE) then b <- $CPL2_n$(b)

if (¬γ = TRUE) then c <- $CPL2_n$(c).

12. (Currently Amended) An electronic component comprising calculation means programmed to implement a method according to ~~one of Claims 1 to 11, the~~ claim 1, said calculation means comprising ~~in particular~~ a central unit associated with a memory comprising several registers for storing the data a and b.

13. (Currently Amended) A chip card comprising an ~~integrated circuit~~ electronic component according to Claim 12.